INTERNET DOCUMENT INFORMATION FORM

A . Report Title: An Empirical Study of TCP/IP Performance Over ATM

B. DATE Report Downloaded From the Internet 9/22/98

C. Report's Point of Contact: (Name, Organization, Address, Office Symbol, & Ph #):

Nasa Lewis Research Center
21000 Brookpark Road
Cleveland, OH 44135-3127
ATTN: Doug Hoder (216) 433-8705

D. Currently Applicable Classification Level: Unclassified

E. Distribution Statement A: Approved for Public Release

F. The foregoing information was compiled and provided by:
DTIC-OCA, Initials: VM___ **Preparation Date**:_9/23/98________

The foregoing information should exactly correspond to the Title, Report Number, and the Date on the accompanying report document. If there are mismatches, or other questions, contact the above OCA Representative for resolution.

An Empirical Study of TCP/IP Performance Over ATM

I. Sebüktekin[1], T. Bogovic[2] and P. Krishnaswamy[3]
Bellcore
445 South Street, Morristown, NJ 07960[4]

Abstract

This paper outlines some basic performance characteristics of the Transmission Control Protocol / Internet Protocol (TCP/IP) over Asynchronous Transfer Mode (ATM). It also discusses a few solutions to assure acceptable TCP/IP performance over ATM which are implemented by the industry the during the last couple of years. The conclusions in this paper are based on empirical TCP/IP performance test results collected on a DS3 ATM research testbed, architected with commercially available IP and ATM equipment.

TCP/IP performance can vary widely and suffer significantly over ATM networks with large Bandwidth*Delay products. First, it is essential that the TCP window size matches the Bandwidth*Delay product of the end-to-end connection to fully utilize the bandwidth provided by the broadband network. Even if the window size meets this criteria, TCP performance can still be unacceptable, especially if the buffering within the ATM network is limited. A single limited-buffer bottleneck is sufficient to degrade the performance of a TCP connection when multiple traffic sources congest the bottleneck resource, such as in ATM networks with small buffer switches. One approach to assure acceptable TCP/IP performance is to limit the data rate into the bottleneck resource by exercising rate control at the entry to the ATM network. A better solution is to provide sufficient buffering within the ATM network.

1 Introduction

Asynchronous Transfer Mode (ATM) [1] is a scalable broadband switching and transport technology which has the capability to switch and transport multimedia information (i.e.voice, data, and video) simultaneously. It is being considered by many segments of the computing and

[1] (201) 829-4725, isil@bellcore.com, MCC 1C-258B
[2] (201) 829-4348, tjb@bellcore.com, MCC 1G-233B
[3] (201) 829-3048, kri@bellcore.com, MCC 1C-213B
[4] No specific equipment is being endorsed. Tests were performed with available equipment in the lab.

communications industries to support current as well as future broadband services.

To allow ATM and other networking technologies to be combined to become overall end-to-end services, it is necessary that the Internet Protocol (IP) [2] operate over ATM just as today's Internet operates over a wide range of different underlying network technologies. Using IP as the common denominator, the approach accommodates an increasingly diverse range of end-to-end applications such as enterprise networking and multimedia communications in a heterogeneous networking environment. To support this objective, one must ensure that TCP [3] /IP performs well over ATM network substrates.

Being new, and quite different from the widely-used packet switching technologies based on broadcast-media (i.e. Ethernet, Token Ring, FDDI), ATM has experienced some lackluster support in some quarters of the industry in contrast to enthusiastic support from others. As a maturing technology, problems inevitably exist such as lack of fully developed standards, unoptimized performance, and not fully established, but evolving equipment. However, over the last few years, ATM has rapidly matured and most importantly has been enjoying widespread support for standards relating to IP. There are currently deployed, operational ATM networks exhibiting increased IP connectivity and traffic, reflecting the higher bandwidth demands of the networking community.

This paper summarizes some performance characteristics of TCP/IP over ATM, and discusses some solutions offered in the industry towards improving TCP/IP performance over ATM. Experimental results are provided to support the main ideas presented. Although performance results over a DS3 ATM network are provided here, similar results and conclusions can be extrapolated for an OC3 ATM network. The results are presented in the form of performance curves plotting end-to-end TCP throughput versus maximum offered TCP window sizes.

Section 2 of this paper describes the DS3 ATM testbed architecture and some specifics relevant to the performance tests and the test tools used. Section 3 briefly discusses the performance impact of protocol overheads, and Section 4 presents baseline performance results measured over the ATM testbed. Next, based on experimental results, Section 5 describes the impact of congestion, on TCP/IP performance, at a DS3 ATM bottleneck with limited buffering and Section 6 examines two industry-implemented solutions to rectify the resulting degradation in performance. Finally, Section 7 summarizes the conclusions of this work.

2 Testbed Architecture

Figure 1 presents the architecture of the TCP/IP over DS3 ATM research testbed. It consists of a single ATM switch, three IP routers with direct DS3 ATM UNI [4] access to the switch and three high performance workstations, SGI Indigo2s, used as TCP/IP data sources and sinks.

Each workstation is on a dedicated FDDI ring and has the capability to source and sink data at almost the full FDDI rate of 100Mbps. These workstations provide the desired data loading capabilities for the DS3 ATM UNIs in addition to two important capabilities necessary for efficient utilization and testing of large Bandwidth*Delay product networks: 1) Large TCP window sizes, and 2) MTU path discovery. The Indigo2s run the SGI's Unix Operating System Irix 5.3 which implements RFC 1323 [5], [6], the TCP extension that allows for extended TCP window sizes. Irix 5.3 allows TCP window sizes up to 524,288 Bytes[5] - sufficient to fully load and test DS3 (45Mbps) networks with Round Trip Delays (RTDs) up to 93 msec. Irix 5.3 also implements MTU path

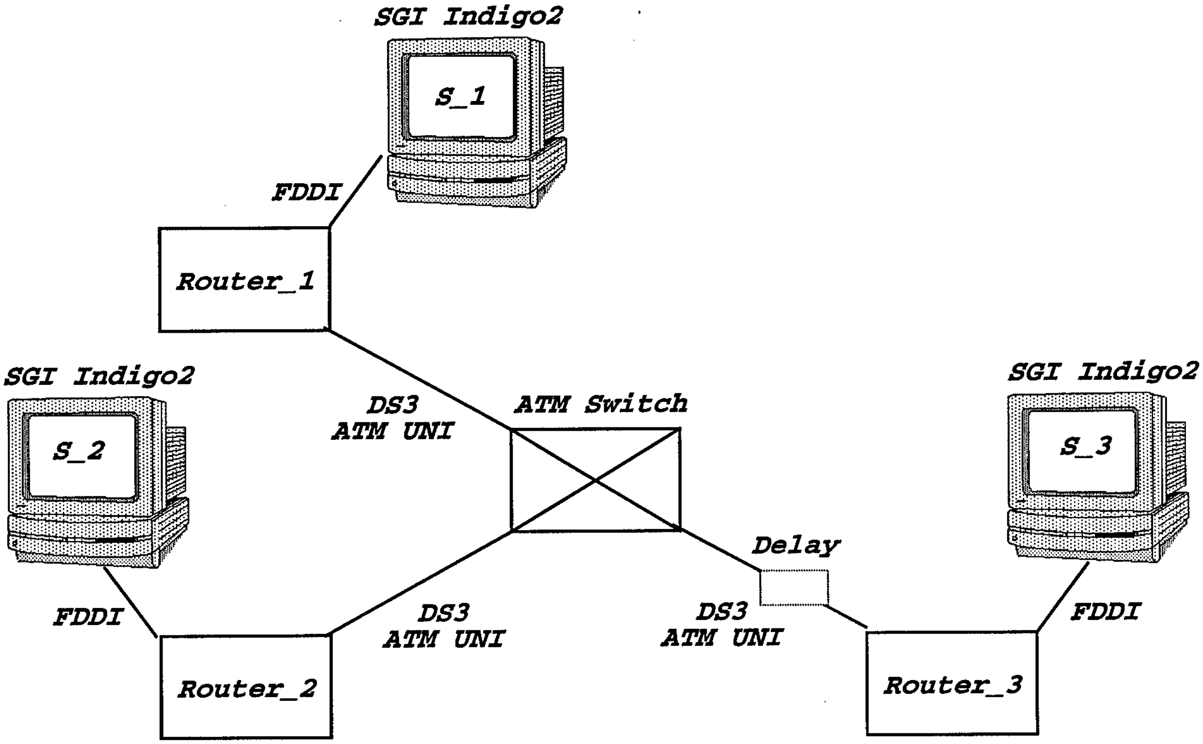

Figure 1 TCP/IP Over DS3 ATM Testbed

[5] Unix BSD4.3 supports up to 64 Kilobyte TCP windows. 16 Kilobyte windows are the most common.

discovery [7] and sets the TCP default maximum segment size (mss) to the minimum Maximum Transmission Unit (MTU) in the network (minus the TCP/IP header)[6]. In Figure 1, the FDDI MTU of 4352 Bytes (minus the 40 Bytes TCP/IP header) is used as the maximum TCP segment size. It is a well-known networking phenomenon to use the largest packet size possible while avoiding packet fragmentation, in order to reduce the per packet overhead at the hosts, and to attain the maximum throughput through the network [8], [9], [10], [11]. The TCP version implemented in the Irix 5.3 kernel is 4.3BSD/Reno that provides the Fast Retransmit and Fast Recovery algorithms [12] which are modifications to the original TCP congestion avoidance algorithm [13].

In the TCP/IP over DS3 ATM testbed displayed in Figure 1, two different ATM switches are used; one switch with a small number of cell buffers per port (on the order of a few hundreds) and the other with considerably larger cell buffers (on the order of a few tens of thousands). The ratio between the per port cell buffers of the two ATM switches is on the order of 100/1. Under full load, the two switches provide the TCP flows with significantly different performance characteristics as will be explained in Section 5. The traffic service tested on the ATM switches is the Unspecified Bit Rate (UBR); i.e. no traffic shaping or policing has been provided at either one.

The three IP routers used in the experiments have ATM interfaces that provide direct DS3 ATM UNI access to the ATM switches. They have the capability to rate shape the data into the ATM network, either using Constant Bit Rate (CBR) or Variable Bit Rate (VBR) as defined by the ATM Forum [14]. In the absence of commercial Available Bit Rate (ABR) implementations, rate shaping ingress to the ATM network provides performance gains for UBR switches with limited buffers, as will be demonstrated in detail in Section 6.1. Rate shaping can alternatively be exercised at the source workstations by adjusting the inter-packet spacing. Network interface cards that perform rate shaping by varying the inter-packet gaps are available in the market for various workstations. The testbed used in this study provides rate shaping at the entry to the ATM network because of the type and availability of equipment employed at the time, it was not a matter of preference.

An Adtech SX/13 Data Channel Simulator with DS3 interface was used to introduce additional network delay in certain tests. It has the capability of introducing RTDs up to 100 msec. The Data Channel Simulator, or Delay Generator, for testing purposes, is connected between the ATM

[6] TCP segment is the payload of the TCP packet. The joint TCP/IP header is 40 Bytes (without any options)

switch and Router_3 in Figure 1, since Router_3 is used as the sink router (thus S_3 as the sink station) in most of the test studies. In this way, the same network delay can be introduced for both sources, S_1 and S_2, when two-to-one tests (two sources sending to the same sink) are conducted.

2 • 1 Performance Tests and Test Tools - A Summary

The performance tests carried out over the ATM testbed displayed in Figure 1 are either single-flow (one-to-one) or dual-flow (two-to-one) TCP performance tests, as permitted by the testbed architecture. These are throughput measurements of large TCP flows (single or dual) during each of which the maximum TCP window size offered is varied. In other words, each test corresponds to a long TCP session for which the throughput is measured when the maximum offered TCP window size is set to a certain value. The enforced window sizes range from 16 to 524 Kilobytes in 16 Kilobyte increments. For a maximum window size set, each TCP session lasts until 81,920,000 Bytes (10,000 8192-Byte application PDUs (TCP payload)) are transferred which is sufficiently long to collect accurate measurements for the range of window sizes used. The measured TCP throughput values are plotted versus the range of maximum offered TCP window sizes, for each configuration under test, as presented in the rest of this paper.

The throughput versus TCP window size measurements presented here were performed using a Bellcore shell script (written by Grenville Armitage) called **ttcp-multi**, which in turn uses a public domain performance tool, called **ttcp** [15]. ttcp-multi automates multiple consecutive ttcp's by incrementing the window size offered to each ttcp, checks the network statistics for TCP timeouts and retransmissions for each session, and post-processes the results for data presentation. The same set of tests were later repeated while **tcpdump** [16] was running in parallel to ttcp-multi in order to find out the exact details of data transmission for these TCP flows.

During these tests, the Nagle algorithm [6] was disabled by setting the TCP_NODELAY [15] option to push data out to the link as fast as possible since these are bulk data transmission measurements. The Don't Fragment (DF) flag in the IP header was set on all the packets and Push (P) flag in the TCP header was set on every other packet, as observed by the tcpdump [16] output that ran in parallel to the repeated ttcp-multi tests. The tcpdump output also identified that 4096 Bytes of data were being sent in each data packet, thus not really utilizing the maximum TCP segment size of 4312 Bytes. What was happening was the first 4096 Bytes of each application PDU was being pushed into the link (P flag set) having disabled the Nagle algorithm [6], and the next 4096 Bytes were sent out in the next TCP packet. For window sizes less than 61440 Bytes, the size of each TCP data packet was exactly 4136 (4096+40) Bytes, otherwise it was 4148 (4096+40+12) Bytes due to the 10-Byte TCP timestamp option added by the RFC 1323 implementation for extended window sizes, and the two 1-Byte No Operation (nop) options used to align the TCP packet in 4-Byte boundaries necessary for the 32-bit or 64-bit machines. The 3-Byte window scale option is only carried in the SYN packets of the connection establishment phase; not in the data packets themselves. So is the 4-Byte maximum segment size (mss) option which indicated 4312 Bytes.

3 Effects of Protocol Overheads

ATM has been shown to be inefficient for IP datagram services [17] due to various overheads:

- PLCP[7] formatting (on average, the DS3 PLCP transmits a cell every 10.42 microseconds),
- 5-Byte ATM cell header for each cell payload formed by segmenting the AAL5[8] packet,
- 8-Byte LLC/SNAP[9] header [18] encapsulating IP in AAL5 [19] and 8-Byte AAL5 trailer, and
- Any AAL5 padding, necessary to divide the AAL5 packet evenly into 48-Byte cell payloads.

Theoretically, the maximum cell payload throughput over a DS3 ATM network is 36.86 Mbps ((53*8/10.42)*48/53). This is a total of 18% overhead just due to the ATM cell headers and PLCP framing. Since, the maximum IP packet that can be used in the testbed in Figure 1 is 4352 Bytes (FDDI MTU) as explained in the previous section, it corresponds to an AAL5 packet of 4368 Bytes (4368/48=91 cells without any AAL5 padding). Therefore, the maximum IP throughput is calculated as 36.72 Mbps (36.86*4352/4368). When the TCP/IP headers are also considered, the TCP data throughput falls further down to 36.39 Mbps (36.72*4312/4352), yielding a total of 19% overhead. The actual maximum TCP data throughput measured over the DS3 ATM testbed in Figure 1 is slightly less than this value calculated. This, of minor concern, is due to several reasons such as:

- varying packet processing delays at the workstations (100 μsec to 5 msec),
- the amount of AAL5 padding inserted in the TCP packets to align with cell boundaries,

[7] Physical Layer Convergence Protocol
[8] ATM Adaptation Layer 5
[9] Logical Link Control / SubNetwork Attachment Point

- the TCP/IP header being actually larger than 40 Bytes due to the TCP timestamp option RFC1323 implementations use with data packets, for extended TCP windows, (larger SYN headers due to options used with the SYN packets can be neglected since the data transfer phase is very long)
- data packets being slightly smaller than the actual TCP maximum segment size negotiated during MTU path discovery.

If the above theorethical calculations are repeated taking into account these reasons listed above, calculations yield very close values to the throughput values observed in the actual measurements. For example for the window sizes less than 61440 Bytes, the data payload of 4096 Bytes (as tcpdump outputs reveal) means a TCP segment of 4136 Bytes, and an AAL5 PDU of 4176 Bytes (87 cells) including some padding. This yields a TCP data throughput of 36.15 Mbps (36.86*4096/4176) and when the maximum segment size underutilization effect is taken into account, it is further reduced to 34.36 Mbps (36.15*4136/4352). For larger window sizes, the TCP segment is 12 Bytes larger, 4148 Bytes, and the same calculations yield 34.46 Mbps.

4 Baseline TCP/IP over ATM Performance Tests

Figure 2 and Figure 3 present two baseline performance tests for single-flow TCPs (from a single source workstation to a sink workstation, i.e. from S_1 to S_3): first, through the small buffer ATM switch, and second, through the large buffer ATM switch respectively, in the testbed configuration in Figure 1. Both of these tests are performed under the exact same conditions; at full DS3 rates when no rate shaping is exercised by the source IP router (Router_1) at the ingress to the ATM network and when no additional network delay is introduced by the delay generator.

These one-to-one or single-flow tests are important since they form a baseline for the following two-to-one tests which exhibit congestion. Full-load two-to-one tests cannot perform any better than these single-flow tests. The one-to-one tests characterize the amount of buffering at the ingress IP router, since there is no slower link ahead. The two-to-one tests are performed to assess the ATM switch behavior under congestion and to characterize the buffering in the switch which becomes the bottleneck resource for more than one flows.

The following observations hold for both of these tests. RTD is around 1 msec and the packet processing delays range from 100 µsec to 5 msec, which contribute to the overall delay. Thus, the Bandwidth*Delay Product is small compared to larger RTD networks and full utilization is achieved at smaller window sizes resulting in the steep upward slope on the graph. The maximum observed throughput of 34-34.5 Mbps is sustained beyond 400 Kilobyte windows. The roughly 400 Kilobyte width of the high throughput section of the graph indicates the amount buffering in the source router (Router_1 in this case, but all the routers are identical). Since a single-flow TCP session does not create a bottleneck over the ATM switch, the buffering in the ATM switch does not have a significance in these single-flow tests, as indicated with like performance with either of the switches (Figure 2 and Figure 3). The throughput collapses beyond 400 Kilobytes due to several packet drops at the source IP router (overrun buffers) and subsequent TCP retransmissions. Both performance tests show throughput settling at similar throughput

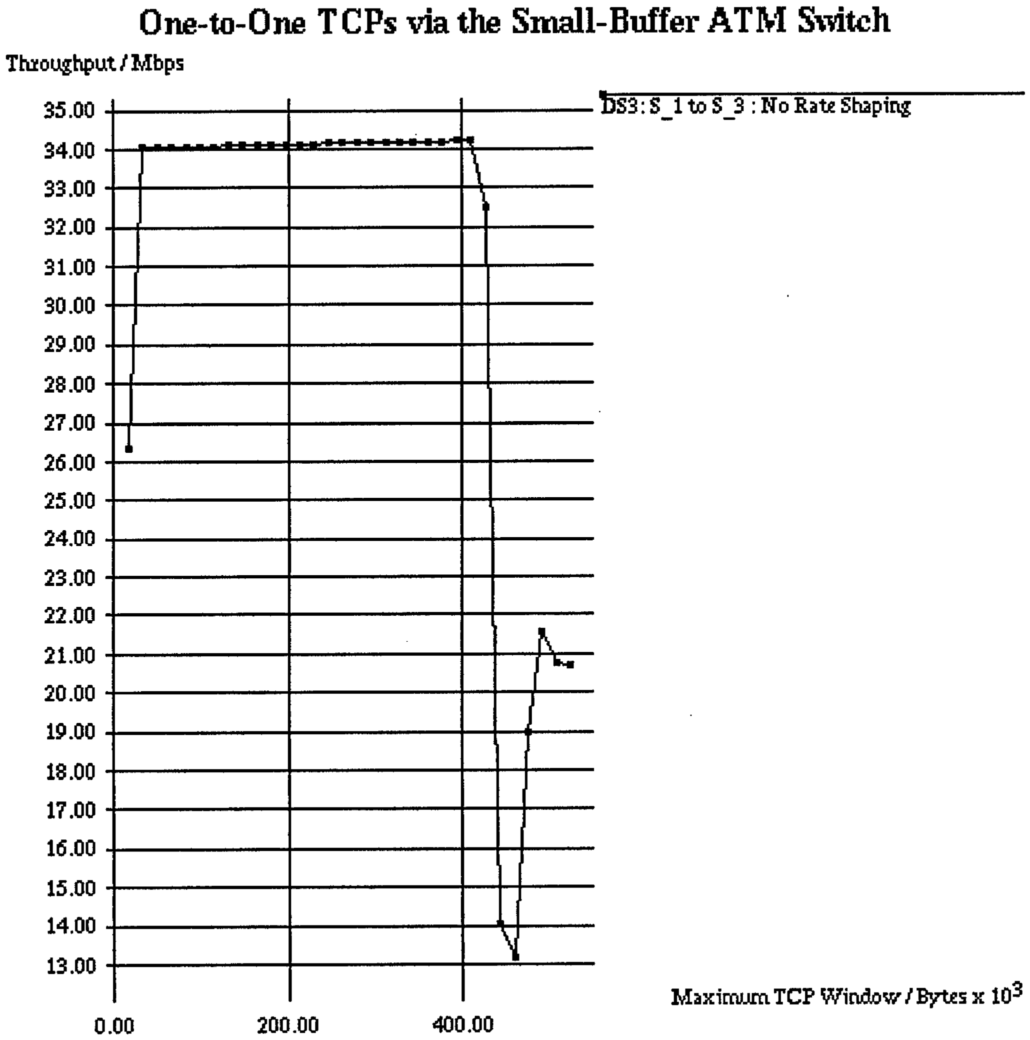

Figure 2 Performance Tests for Single-Flow TCPs Through the Small Buffer ATM Switch with no Rate Control Exercised at the Entry to the ATM Network (RTD = 1 msec)

values, around 17-18Mbps beyond the collapse point. The glitch in Figure 3, around 260 Kilobytes is due to a few TCP retransmissions that occurred during that specific TCP session. However, such glitches due to a series of loss, timeout, retransmission events that degrade the performance are completely random, in other words the same glitch does not reoccur at different repetitions of the same test; such glitches are infrequent under no congestion cases.

4 • 1 Effects of Additional Delay on Performance

Figure 4 displays another single-flow TCP test through the small buffer ATM switch, with an additional 20msec delay introduced in the data path, as opposed to the test displayed in Figure 2.

The objective of adding network delay, is to emulate high latency networks, thus to test long

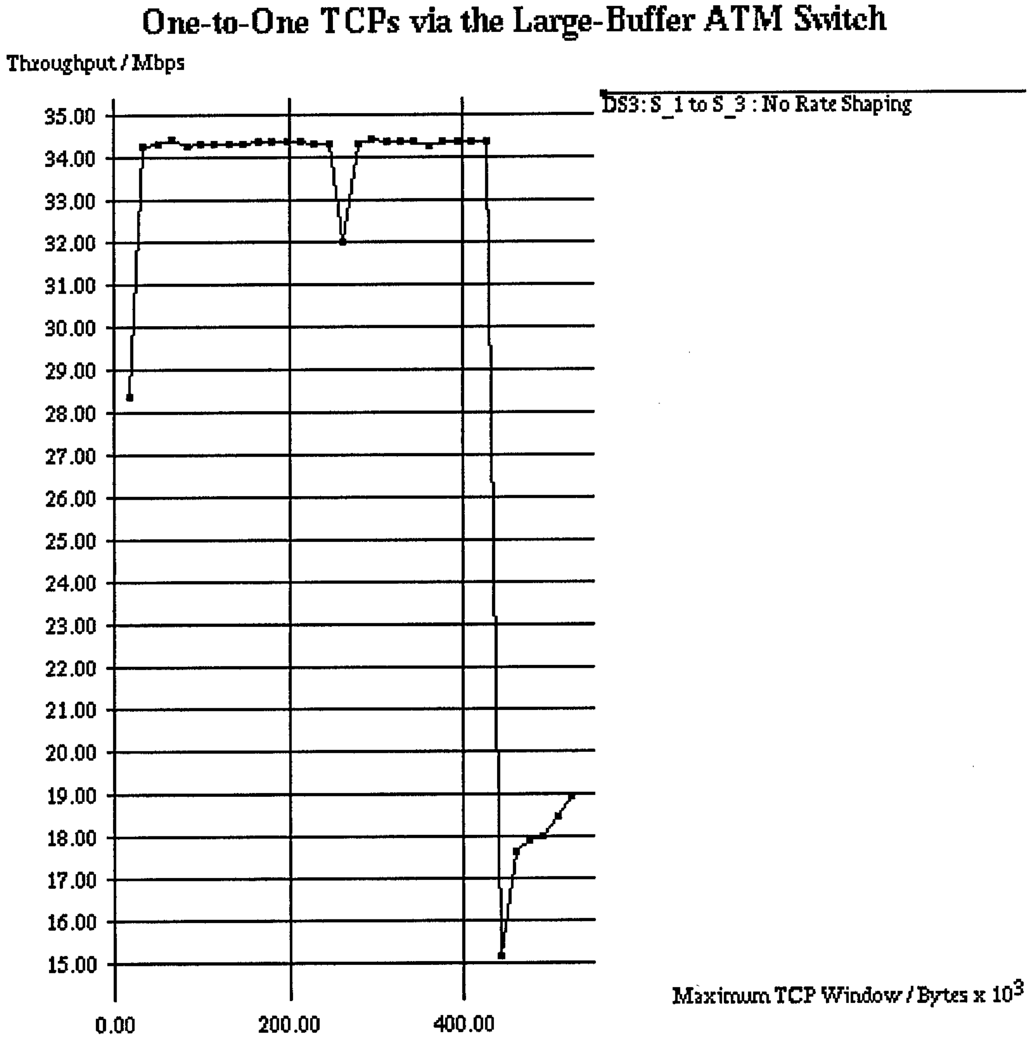

Figure 3 Performance Tests for Single-Flow TCPs Through the Large Buffer ATM Switch with no Rate Control Exercised at the Entry to the ATM Network (RTD = 1 msec)

delay paths. With added delay, Bandwidth*Delay product of the data path increases, requiring larger windows to fully utilize the link.The rule of thumb for TCP/IP networks is that 100% link utilization, thus maximum throughput, is achieved when the TCP window size matches the Bandwidth*Delay product of the data path, where Bandwidth is the capacity of the slowest link.

Since the maximum attainable throughput is reached at a higher window size for a larger Bandwidth*Delay product path, the effect of additional delay is a smaller slope on the throughput versus window size graph in Figure 4 compared to that in Figure 2. However, the amount of buffering within the physical equipment over the data path is independent of the network delay, and the width of the high throughput section is the same on both of the graphs. In fact, this width

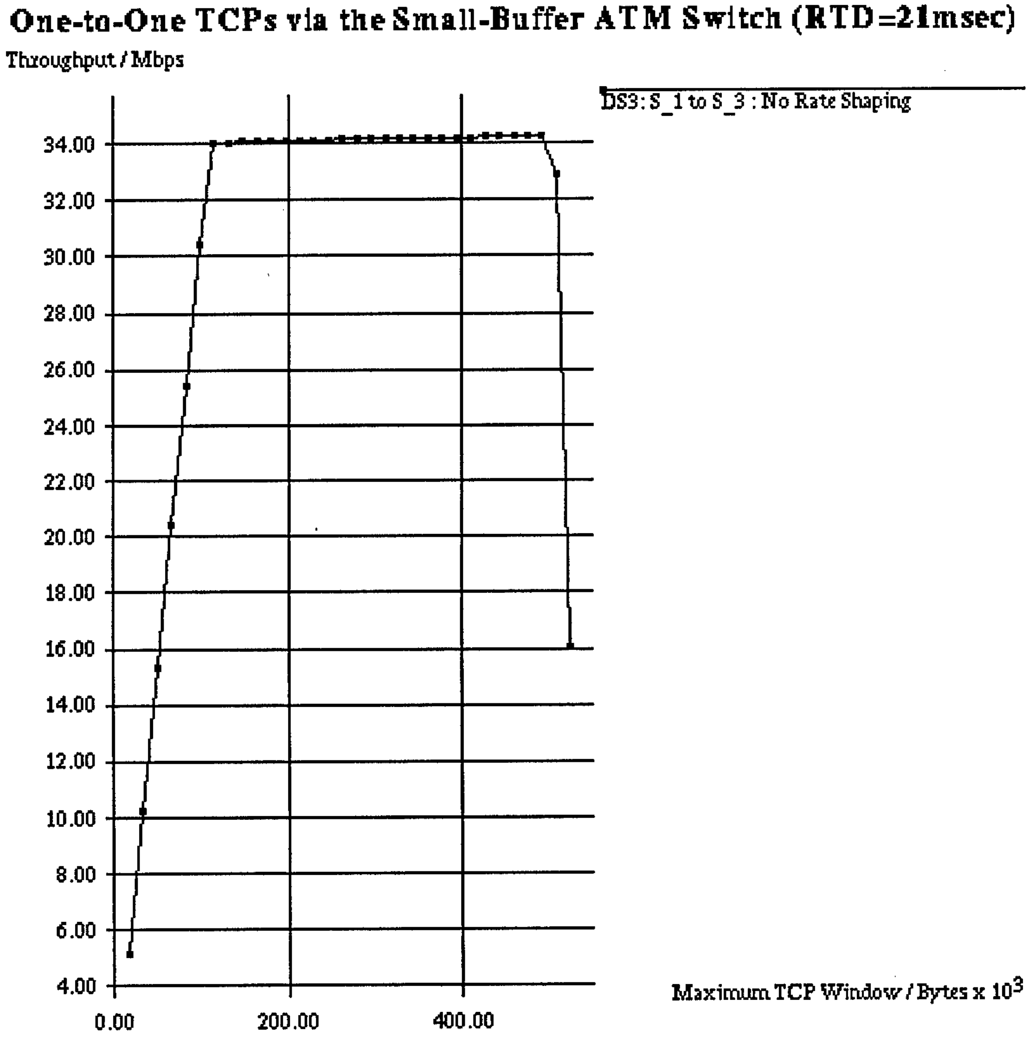

Figure 4 Performance Tests for Single-Flow TCPs Through the Small Buffer ATM Switch with no Rate Control Exercised at the Entry to the ATM Network (RTD = 21 msec)

directly indicates the amount of bottleneck buffering (i.e. buffering in the ingress router) over the data path. Consequently, additional delay also shifts the throughput collapse point, i.e. the window size at which the bottleneck resource overflows its buffers and degrades the throughput.

In summary, with a higher RTD, the window sizes at which maximum throughput is reached, and throughput collapses, are larger than those with a lower RTD.

5 Impact of Congestion

Unlike the results presented in Figures 2,3 and 4, TCP performance can vary widely and suffer significantly if multiple traffic sources congest a bottleneck link with limited buffering. Figure 5

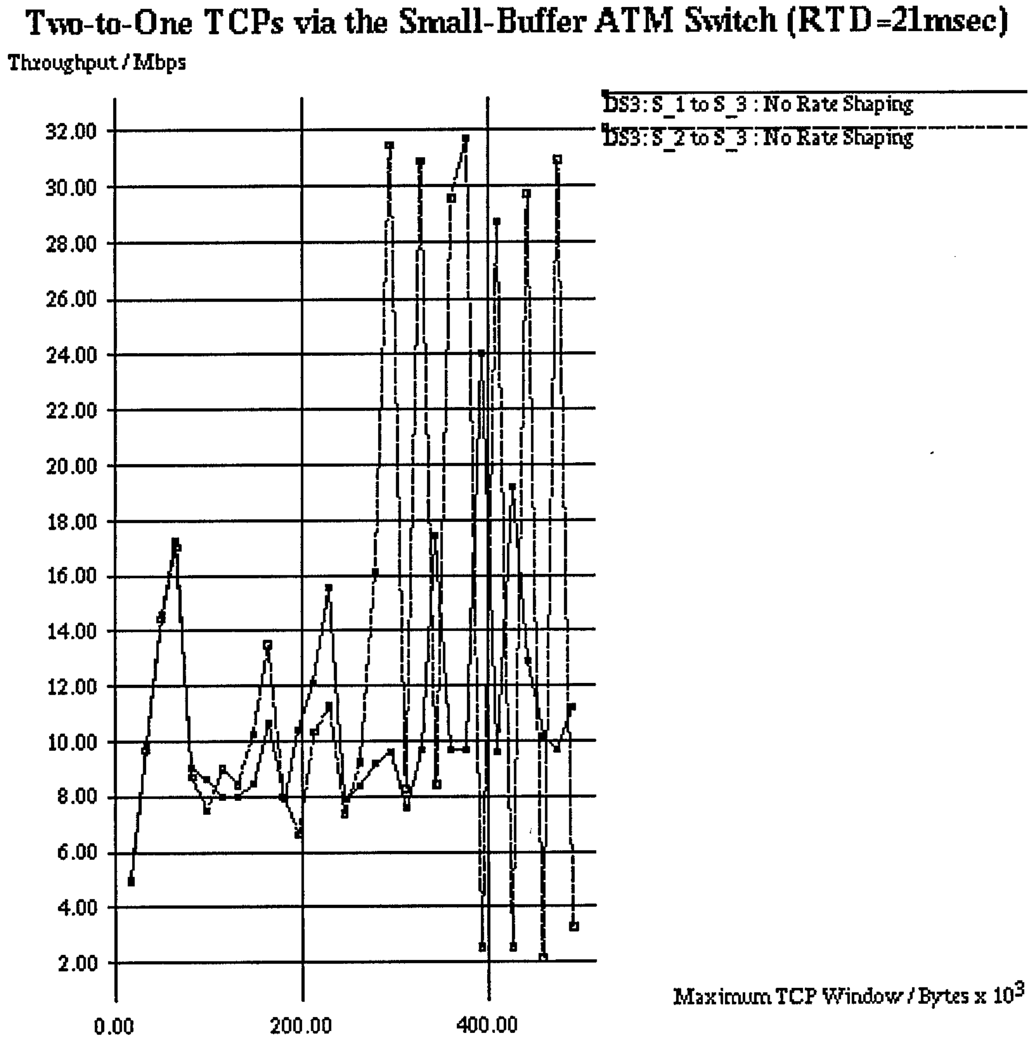

Figure 5 Performance Tests for Two-to-One TCPs Through the Small Buffer ATM Switch with No Rate Control Exercised at the Entry to the ATM Network (RTD = 21 msec)

displays the results of such a performance test in which two TCP sources simultaneously send to the same sink (i.e. from S_1 and S_2 to S_3) through the small buffer ATM switch, when there is no rate control exercised at either of the source routers (Router_1 and Router_2). In this test, two DS3 sources at full rate, congest the outgoing DS3 ATM port (towards Router_3) over the switch, and since the switch has limited cell buffering, it starts to drop cells from either one of the flows. Which TCP session is losing its cells is a random behavior, and performance results display various throughput degradation effects as will be explained in detail.

The performance test displayed in Figure 5 is performed using an additional 20msec delay introduced by the Delay Generator to more clearly depict the initial and acceptable performance characteristics, i.e. the initial slope where the throughput increases as the window sizes increase. During this time, both flows exhibit equal throughput values as the share of bandwidth assigned to each is also equal. They both reach a throughput of approximately 17.25Mbps (aggregate throughput is 34.5Mbps) at a window size approximately half the Bandwidth*Delay product (The two window sizes together are equal to the Bandwidth*Delay product). Thus, once the window sizes of both flows become large enough to fill the outgoing DS3 pipe and the small buffering in the ATM switch runs out, both flows start to lose cells at a high rate. Thus, both flows' congestion control mechanisms [8], [13] commence and both switch into the slow start state quickly. Many cycles of slow start/congestion avoidance repeat until both flows finally finish the data transfer process. This is the synchronization artifact of TCP. However, as the TCP window sizes used become larger than 200 Kilobytes, a different artifact of TCP becomes apparent, and that is domination of one flow over the other. At such high window sizes, the burst of data arriving from either of the flows is much larger, and in most cases, the flow ahead of the queue manages to get through while the other one loses a lot of cells. Once that occurs, the leading flow dominates the other which has backed off and continues to back off until the dominating one finishes its session. Only then can the backed-off flow proceed faster; however, due to the difficulty in raising its congestion window much higher, it displays dismal throughput, much smaller than half the throughput of the other flow. This unfair behavior can work against either flow[10]; it is completely random, as exhibited in Figure 5; at certain window sizes, the S_1 originated flow dominates,

[10] Domination effects are not clear on black-and-white. Authors can provide color graphs, if requested.

while during other times, the S_2 originated flow dominates.

6 Possible Solutions

The congestion-induced degradation in performance (due to synchronization) or unfair bandwidth utilization (due to domination) can be rectified in a number of ways. This section examines two possible solutions - source rate control and increased switch buffering - based on experimental results. These solutions are by no means the only methods that exist to overcome these problems. For example, there are ATM switches available in the market that implement Packet Discard algorithms (Early Packet Discard (EPD) or Partial Packet Discard (PPD)) discussed in various literature [20], [21]. There are ongoing experimental studies of these methods within Bellcore which are not available for publication at the present time. ABR switches will be able to address these problems as well, however they are just becoming available in the market.

6 • 1 Rate Control

One way to assure acceptable TCP/IP performance for ATM networks with limited buffer switches is to control the amount of data into the ATM network, thus into the bottleneck resource. This can be provided by exercising rate control at the ingress to the ATM network, either at the data sources or the IP routers interfacing to the ATM network. As long as either provide adequate rate control and buffering, either should be effective. Experiments here are carried out using rate shaping at the IP routers. Rate shaping corrects the undesirable behaviors explained in Section 5.

Figure 6 presents the results of such a performance test using the small buffer ATM switch during which CBR shaping is exercised at both source routers (Router_1 and Router_2). The bandwidth assigned to each flow is limited to 20Mbps in this test, and an additional delay of 20msec is introduced between the ATM switch and the sink router (Router_3) for direct comparison with the results displayed in Figure 5.

Once the flows reach 16.5-17 Mbps maximum at the same window size as in Figure 5, they both sustain that level until the buffers on their source routers (Router_1 and Router_2) start to overflow. Note that the width of the high throughput section of the graph in Figure 6 is equal to those of the graphs in Figures 2, 3, and 4 and is indicative of the amount of buffering provided by the ATM interfaces of the ingress routers as indicated earlier. The glitch in Figure 6 is also due to a random sequence of loss, timeout and retransmission events as for the glitch in Figure 3.

Figure 7 presents the results of another similar performance test using the small buffer ATM switch. The difference between Figure 6 and Figure 7 is that Figure 7 depicts VBR shaping at each of the source routers (Router_1 and Router_2) and no additional delay is provided. The average bandwidth assigned to each flow is still 20Mbps, however the peak rate is set at 40Mbps. A small burst size equal to the FDDI MTU is used in this test. Since the 20msec additional delay is no longer used, maximum throughput is reached at very small window sizes (as in Figure 2 and Figure 3). Due to the VBR shaping, the throughput slightly increases as higher window sizes are used, unlike the flat throughput observed in Figure 6. The maximum throughput reaches 17-17.5

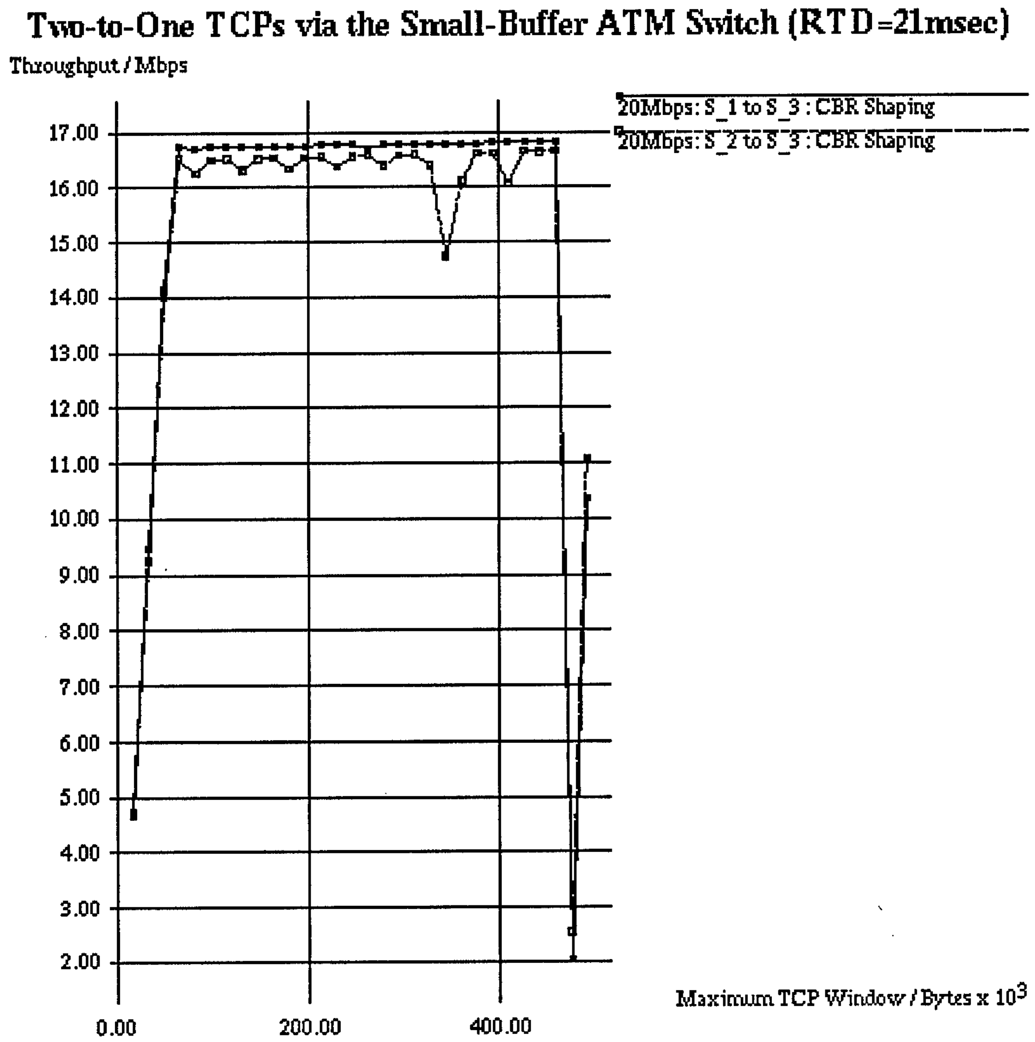

Figure 6 Performance Tests for Two-to-One TCPs Through the Small Buffer ATM Switch with CBR Shaping Exercised at the Entry to the ATM Network (RTD = 21 msec)

Mbps for each flow, thus an aggregate of 34.5-35 Mbps which confirms the calculations in Section 3. Note that the buffering within the routers is the same (400 Kilobyte).

As Figure 6 and Figure 7 demonstrate, the performance gain due to VBR shaping is not significant compared to CBR shaping with the use of small burst sizes. Unfortunately, large burst sizes are found to yield significant throughput degradation when multiple flows congest the small-buffer ATM switch, by sending large bursts of data simultaneously to the DS3 ATM bottleneck. This is expected since the peak rates are 40Mbps each. Unfortunately, the smallest peak rate to average rate ratio that can be set is 2 (a high oversubscription rate) due to the limitations of the commercial equipment available to this study at the time. A capability to choose

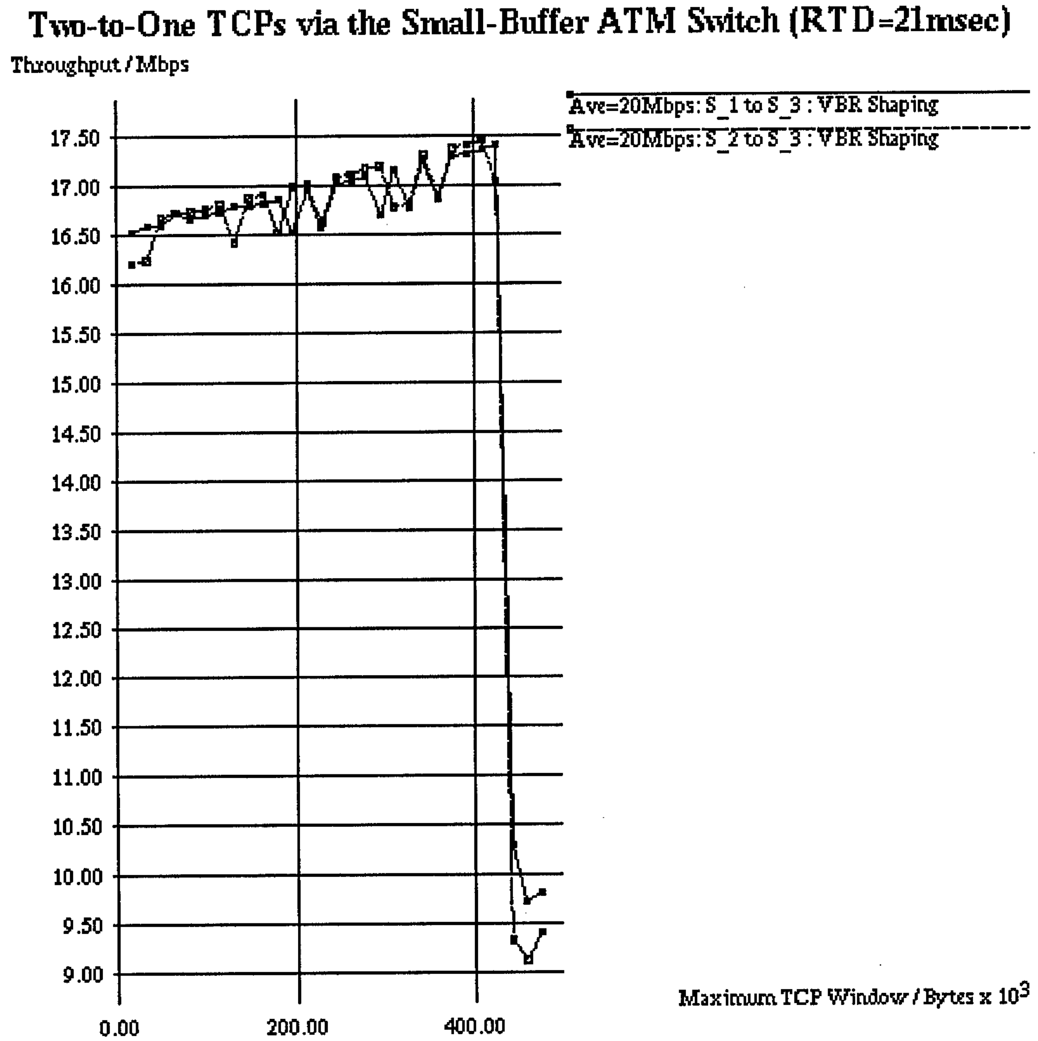

Figure 7 Performance Tests for Two-to-One TCPs Through the Small Buffer ATM Switch with VBR Shaping Exercised at the Entry to the ATM Network (RTD = 1 msec)

peak rates that yield smaller peak rate to average rate ratios should bring forth the advantages of leaky buckets for small-buffer ATM switches, and can then provide statistical multiplexing and improved performance with the use of larger burst sizes. Unfortunately, current inflexible implementations of the algorithm by the commercial equipment used does not help, and is potentially harmful under full load, for small-buffer ATM switches.

Figure 6 and Figure 7 display the cases when the bandwidth assigned to each flow is equal. When the ratio of the bandwidths is not equal to one, the same behavior is observed except that the ratio of each flow's throughput is in proportion to the ratio of the bandwidths assigned.

6 • 2 Sufficient Buffering

Although rate shaping the streams at the entry to the ATM network is an effective solution, it requires provisioning parts of the total bandwidth for specific connections ahead of time (using Permanent Virtual Circuits (PVCs)). Since all the connections, for each of which a portion of the bandwidth is reserved, may not be active at all times, the active connections cannot utilize the unused bandwidth reserved for the inactive ones with CBR provisioning. Furthermore, VBR provisioning implemented in the IP routers is ineffective for small-buffer ATM switches. In addition, provisioning connections requires manual intervention. Switched Virtual Circuits (SVCs) will address these problems, however they are not available for the tests presented here.

A second solution is to provide sufficient buffering at the bottleneck resources (in relation to the Bandwidth*Delay product of the network). Sufficient buffering is a relative term and there is not a single optimum number for best performance under all circumstances. However, it is possible to estimate a range to cover a range of possible RTDs for a given bandwidth. This section attempts to do this by comparing the results of the two tests displayed in Figure 5 and Figure 8. Both of these tests are carried out at full DS3 rates with no rate control mechanism being exercised at the source IP routers (Router_1 and Router_2). The major difference is that the performance curve in Figure 5 is collected through the small buffer ATM switch, and the curve in Figure 8 is collected through the large buffer ATM switch. The ratio of the buffer sizes is about 1:100. The small buffer switch has only a few hundreds of cell buffers per port whereas the large buffer switch has a few tens of thousands of cells per port. There was also no added network delay in the test whose results are presented in Figure 8.

The Bandwidth*Delay product for a 21msec DS3 network is approximately 2000 cells. Clearly, a switch buffer of a few hundred cells per port, less than the Bandwidth*Delay product, provides dismal performance. For example, the throughput in Figure 4 exhibits significant degradation at window sizes immediately after the window size at which it reaches its maximum of 17.5Mbps for both flows. On the other hand, a switch buffer of tens of thousands of cells per port performs well for a wide range of TCP window sizes as the throughput in Figure 7 is sustained at its maximum (around 17.5 Mbps for each of the flows) for a wide range of window sizes, almost up to 350 KiloByte windows per flow. The performance in Figure 8 suffers only after the large cell buffers in the switch run out (beyond 350 Kilobytes), and manifests itself in the form of either

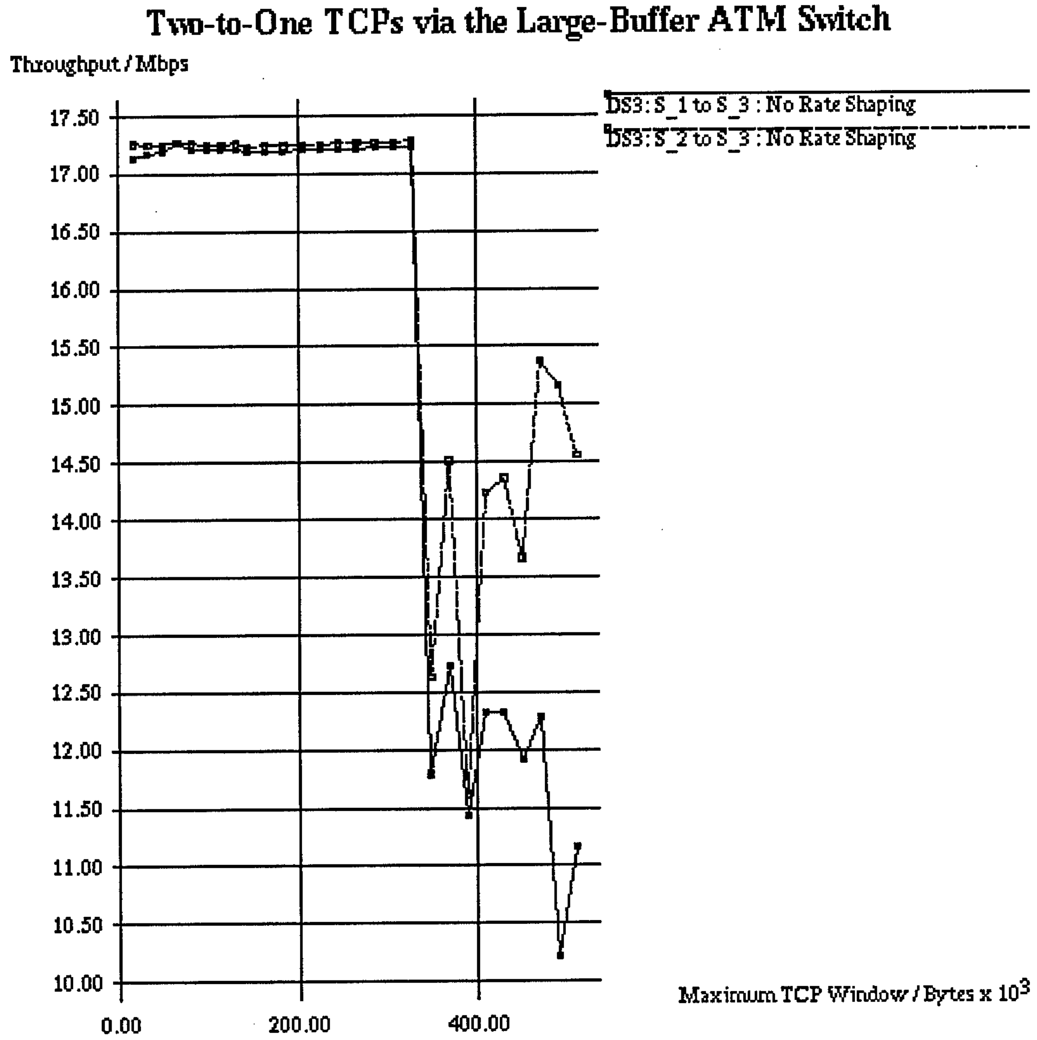

Figure 8 Performance Tests for Two-to-One TCPs Through the Large Buffer ATM Switch with No Rate Control Exercised at the Entry to the ATM Network (RTD = 1 msec)

domination (unfair bandwidth utilization between the two flows) or synchronization (poor performance for both flows), just as in Figure 5. Figure 5 displayed these undesirable performance effects immediately after reaching its maximum throughput due to the small per port buffers being exhausted rapidly. If there was an additional delay of 20 msec, the high throughput section of the graph in Figure 8 would have shifted right, to about 400 Kilobyte windows. No delay was intentionally added to determine the exact per port buffering in the ATM switch.

By comparing the results in Figure 5 and Figure 8, one can see that sufficient cell buffers provide good throughput performance for a wide range of TCP window sizes. A switch buffer of tens of thousands of cells per port can comfortably accommodate up to 100msec RTDs over a DS3 network. Thus, network delays that can possibly change during a TCP session won't likely degrade the performance characteristics of the session. However, with limited cell buffers, a change in RTD during a session may have quite a negative impact on the performance of the session. Fine tuning window size parameters to the Bandwidth*Delay product of the network is required with limited buffering in the ATM network. Such problems do not exist for larger buffer switches, as long as the window size is set larger than the Bandwidth*Delay product of the connection. Larger buffers have been considered and implemented by many ATM switch vendors already. The reduced costs of memory compared to a few years ago make it even a more viable and attractive solution.

7 Conclusion

This paper discusses two methods as solutions to guarantee acceptable TCP/IP performance over an ATM network. State of the art equipment is available in the market to implement either of the solutions.The paper presents experimental results collected over a TCP/IP over ATM research testbed to demonstrate the improved performance using these methods. One method is to provide rate shaping (CBR or VBR or other viable algorithms) at the entry to the ATM network, in proportion to the bandwidth of the bottleneck resources. The other method, which has significant advantages, is to provide ample buffering in the ATM switches. The superiority of the latter to the first is that it does not require provisioning of only a portion of the bandwidth to a PVC, thus the whole bandwidth can be statistically multiplexed among active TCP flows. In theory, the first method also provides statistical multiplexing gain when leaky bucket mechanisms (VBR shaping) are used, however, it requires larger switch buffers as well in order to sustain simultaneous large bursts contending for the bottleneck resources.

There are other methods suggested that would improve TCP/IP performance over ATM further such as Packet Discard algorithms and ABR. Since memory prices have dramatically decreased over the years, adding large buffers to ATM switches is the preferred choice and can be accompanied with any other performance improving method. Adequate buffering allows for increased Cell Delay Variation and also assures better performance for long latency data connections, whether they are cross-country or intercontinental connections that may span satellite links, which have considerably high RTDs.

Acknowledgments

Special thanks to Dave Feldmeier, Ken Young, and Dan Daly for their valuable suggestions on the paper. Also gratitude to Dan Daly for his support in submitting this paper. Finally, many thanks to Grenville Armitage who had written the ttcp-multi test script which greatly simplified the performance measurements during this work.

References

[1] M. De Prycker, "**Asynchronous Transfer Mode, Solution for Broadband ISDN**", Second Edition, Ellis Horwood, 1993.

[2] J. Postel, "Internet Protocol", RFC 791 (Standard), Sept. 1981.

[3] J. Postel, "Transmission Control Protocol", RFC 793 (Standard), Sept. 1981.

[4] ATM User-Network Interface Specification, Version 3.1, ATM Forum, Sept. 1994.

[5] V. Jacobson, R. Braden, and D. Borman, "TCP Extensions for High Performance," RFC 1323 (Proposed Standard, Obsoletes RFC 1185), May 1992.

[6] W. R. Stevens, "**TCP/IP Illustrated, Volume 1 - The Protocols**", Addison-Wesley, 1994.

[7] J. Mogul, and S. Deering, "Path MTU Discovery", RFC 1191 (Draft Standard), Nov. 1990.

[8] C. Villamizar and C. Song, "High Performance TCP in ANSNET", ftp://ftp.ans.net/pub/papers/tcp-performance.ps, Sept. 1994.

[9] D. Clark, V. Jacobson, J. Romkey, and M. Salwen, "An Analysis of TCP Processing Overhead," *IEEE Communications Magazine*, vol. 27, pp. 23-29, June 1989.

[10] D. A. Borman, "Implementing TCP/IP on a Cray Computer," *ACM Computer Communication Review*, vol. 19, pp. 11-15, April 1989.

[11] V. Jacobson, "Some Design Issues for High-Speed Networks," Networkshop'93, Melbourne, Australia, Nov. 1993.

[12] V. Jacobson, "Modified TCP Congestion Avoidance Algorithm" end2end-interest mailing list, ftp://ftp.isi.edu/end2end/end2end-interest-1990.mail, April 1990.

[13] V. Jacobson, "Congestion Avoidance and Control" *ACM Computer Communication Review*, vol. 18, pp. 314-329, Aug. 1988.

[14] ATM Forum, "Traffic Management Specification," Version 4.0, af-95-0013R11, March, 1996.

[15] ttcp (test tcp), Performance Measurement Tool, ftp://ftp.sgi.com/sgi/src/ttcp, 1993.

[16] tcpdump, ftp://ftp.ee.lbl.gov/tcpdump-*.tar.Z, 1994.

[17] G. J. Armitage, and K. M. Adams, "How Inefficient is IP over ATM anyway?", *IEEE Network*, vol. 9, no. 1, pp. 18-27, Jan/Feb. 1995.

[18] J. Heinanen, "Multiprotocol Encapsulation over ATM Adaptation Layer 5", RFC 1483 (Proposed Standard), July 1993.

[19] B-ISDN ATM Adaptation Layer (AAL) Specification, I.363, ITU-T Recom., April 1991.

[20] G. J. Armitage, K. M. Adams, "ATM Adaptation Layer Packet Reassembly during Cell Loss", *IEEE Network*, vol 7, no. 5, pp. 26-35, Sept. 1993.

[21] A. Romanow and S. Floyd, "Dynamics of TCP Traffic over ATM Networks", *ACM Computer Communications Review*, Oct. 1994.